PREDICTING INDIVIDUAL BOOK USE FOR OFF-SITE STORAGE USING DECISION TREES¹

Craig Silverstein² and Stuart M Shieber³

We explore various methods for predicting library book use, as measured by circulation records. Accurate prediction is invaluable when choosing titles to be stored in an off-site location. Previous researchers in this area concluded that past-use information provides by far the most reliable predictor of future use. Because of the computerization of library data, it is now possible not only to reproduce these earlier experiments with a more substantial data set, but also to compare their algorithms with more sophisticated decision methods. We have found that while previous use is indeed an excellent predictor of future use, it can be improved on by combining previous-use information with bibliographic information in a technique that can be customized for individual collections. This has immediate application for libraries that are short on storage space and wish to identify low-demand titles to move to remote storage. For instance, simulations show that the best prediction method we develop, when used as the off-site storage selection method for the Harvard College Library, would have generated only a fifth as many off-site accesses as compared to a method based on previous use.

1 Introduction

It is a commonplace that libraries never have enough room. For instance, Widener Library, the flagship of the Harvard College Library,

1 This article is based on work supported in part by the National Science Foundation under grant 841-9350/92, and in part by an equipment grant from the Harvard University Library. Much of the work was done while Craig Silverstein was at the Division of Applied Sciences, Harvard University. We would like to thank Dale Flecker and Charles Husbands for normalizing and making available the Harvard College Library databases, and Richard De Gennaro, Lawrence Bowles, Dale Flecker, Charles Husbands, Curtis Kendrick, Margo Seltrar, and Michael Smith for their assistance and comments.

2 Computer Science Department, Stanford University, Stanford, California 94025.

3 Division of Applied Sciences, Harvard University, Cambridge, Massachusetts 02138.

[Library Quarterly, vol 65, no 3, pp 266–293]
© 1996 by The University of Chicago. All rights reserved
0024-2519/96/6609-0002$01.00

is comprised of some 4.8 million volumes, whereas the library building itself has space for only 3.5 million of these. Library systems must balance a desire to make titles generally accessible with the often prohibitive cost of increasing the capacity of on-site library locations. The solution to this dilemma was outlined over 250 years ago by Thomas Jefferson, Harvard's second great library benefactor. (The first was John Harvard himself.) In a letter to the College Authorities of Harvard, Jefferson proposed that "If you want room for modern books, it is easy to remove the less useful into a more remote place" [1]. Former Harvard President Charles Eliot echoed the proposal in his well-known address to the American Library Association, recommending "the division of a library into books in use, and books not in use, with different storage methods for the two classes of books" [2].

The traditional solution to the overcrowding problem, then, is to move low-use titles to a relatively distant off-site location, which can be built on less expensive land and can use efficient compact storage techniques. The Harvard College Library has taken this approach using a depository library in Southborough, Massachusetts. At current rates of book acquisition, titles from Widener will not fill Harvard's depository again after 160 years. Thus, the depositary can for all practical purposes solve the problem of inadequate storage. Ancillary benefits of a depository approach include climate control, enhanced security, and improved preservation of fragile books. However, these benefits must be balanced against the great inconveniences libraries and library patrons experience with a depository system, notably the significant delay and cost inherent in retrieving volumes from the depositary.

Given the necessity for off-site storage and its costs, it becomes important to choose those books to be stored off-site carefully. Ideally, books moved off-site would be those with the lowest expected future usage, but future usage is a problematic concept in two ways. First, the future is unknown and can only be predicted approximately, on the basis of available data concerning past usage. Second, even past usage is difficult to measure given the variety of ways in which books are used: circulation, browsing, reference, and so forth. Only for the first of these can comprehensive statistics be automatically accumulated. For this reason, we concentrate on measured circulation statistics as an admittedly approximate gauge of past use. We must be satisfied by previous reports [3–7] that conclude the frequency of in-house use is highly correlated with the frequency of checkouts, though these conclusions are admit-
we could clairvoyantly pick titles to be moved off-site that we know would not be circulated, making our decisions with full knowledge of future circulation patterns, we would expect the hit rate would be reduced to 0 percent until most of the books are moved off-site. The figure verifies this expected performance; the hit rate is 0 percent for the clairvoyant method up until about 70 percent of the titles are stored off-site.

In order to gauge the overall quality of a choice policy, we adopt a measure of how much better it is than the benchmark random policy. By calculating the relative advantage of the choice policy over the random policy, averaged over all off-site percentages, we get the expected advantage over random (EAR), which provides a measure of the quality of a choice policy. The EAR for a choice policy P can be calculated as $1 - \text{(number of hits using policy P)} / \text{(expected number of hits using random policy)}$. By definition, the EAR for the random policy itself should be approximately 0. The EAR for the clairvoyant policy is around .90, which we will state as a percentage: 90 percent. That is, we would expect (given no assumptions about how many titles to store off-site) that the clairvoyant policy engenders only one tenth as many hits off-site as the random policy. The EAR values for the two choice policies (0 and 90 percent) are graphed in the right-hand portion of figure 1. (This portion of the figure also shows the 95 percent confidence interval for each EAR value estimate, as described in Sec. IIIIB.) Although a clairvoyant policy is not implementable—we cannot see into the future—the thought experiment shows that there is tremendous room for improvement over random choice when deciding which books to put in a depository.

A. Summary of Results

In this article, we explore the design of choice policies by examining a large class of such policies, namely, those that can be expressed as decision trees [8], described in Section III. Essentially all past research on the topic of choice policies has worked with policies in this class, allowing

![Graph](image-url)
us to replicate these results—albeit on a much larger scale (Sec. II). On
the basis of these experiments, we replicate previous results, showing
that, overall, past use is the best single predictor of future use. We
demonstrate, however, that in certain commonly occurring cases (when
only a small percentage of books are to be stored off-site), past use is
a worse predictor of future use than LC class or publication date.

However, the availability of large databases of bibliographic and circu-
lation information and the relatively more sophisticated computer re-
sources now available allow us to go well beyond the simple decision
trees previously considered. We can now examine trees with greater
orders of magnitude, using subsamples of much larger scale and variety.
On the basis of these experiments, described in Section IV, (1) we de-
monstrate that the use of additional criteria in predicting future use can
be helpful, though care must be taken, and (2) we develop a practical
choice policy with an EAR of over 73 percent, a significant improve-
ment in predictive power over previous methods, with EARs of 45–60 percent.

B. Some Methodological Caveats
Before describing our experiments, we digress to mention several limi-
tations of our study, which are shared by earlier research on the same
topic. Some technical limitations are described in Section II.B.

First, as mentioned above, we use circulation as a proxy for, and
approximation of, general use. Though circulation seems to be highly
Correlated with in-house use, the large quantity of in-house use means
that any error in the correlation corresponds to large amounts of in-
house use for which circulation is a poor predictor. This observation
has been explicitly made by Robert Hayes [6, 7]. Thus, any off-site
storage method in which usage prediction is based solely on circulation
may lead to a great decrease in the utility of the on-site collection. Un-
fortunately, we have no efficient way of collecting reliable statistics on
inhouse use on the scale required for use prediction, so we must be satis-
fied with a rough predictor over none at all.

Second, we attempt to develop a methodology that, when imple-
mented on a specific collection, can help in choosing titles to move into
secondary storage and is more accurate in predicting future use than
methods currently used. Not everyone agrees that future use, even if
accurately predictable, is the appropriate metric on which to base such
a location decision. Two main schools of thought have developed as to
how to pick titles to move off-site. One, which may be titled the “What
readers do want” school, proposes that experts in various fields pick
titles in their field that are the least “worthy” and send them to
remote storage. Proponents of this view hope that because of the work
involved in recalling a title from the depository, casual students of a

PREDICTING BOOK USE FOR OFF-SITE STORAGE

One way to ameliorate both these problems is to combine an auto-
mated procedure with manual oversight. This can be done in a variety
of ways. A manual pre-process stage could specify individual titles or
classes of titles (such as reference materials or other titles used fre-
quently in-house) to be exempted from consideration by the automated
procedure. A manual post-process stage could follow a computerized
method with expert review. For instance, a decision algorithm could be
made to list more titles than need to be moved, and experts could pick
the requisite number of books from the candidate list. In either case,
a good prediction algorithm can ease the burden of deciding on titles
to move to remote storage when a main collection becomes too crowded,
even if it does not eliminate the burden entirely.

II. Previous Research

In order to design a choice policy to minimize the expected use of titles
stored off-site, we must have a model that allows predicting the use of
individual titles. The two problems of predicting book use and designing
choice policies are thus closely related (but see Sec. I.B. for a discussion
of some differences between the two problems).

Note that what is needed is a method of predicting the use of individ-
ual titles; it is not sufficient for our purposes merely to predict aggregate
book use (though it is sufficient for other purposes, such as aiding re-
source allocation). A great deal of previous research in predicting book
use addresses the aggregate prediction problem [6, 7, 12–18], modeling
future use distributions in toto as, for instance, mixtures of Poisson
processes, perhaps incorporating a decay factor to model book aging.6

6. Modeling the effect of aging on our future predictions is, by itself, immaterial to the
time we are concerned with, as we wish merely to predict usage in the near
future. Usage in the more distant future can be predicted on the basis of similar models
built at that later date. That is, as long as we make each prediction on the basis of the
most recent data, we need not concern ourselves with how the prediction fares
as it ages. However, an ancillary effect of aging is quite important for our purposes,
that is, deciding on the size of the window into the past that we use to measure past
use. There is a trade-off between having a long window, which provides more data
and therefore more reliable statistics, and having a short window, which provides data
An obvious method for distinguishing among these low-use titles is to take into account bibliographic information when trying to predict future book use. Fussler and Simon did so, examining combinations of decision criteria for predicting book use. For instance, they ranked titles that had never been checked out on the basis of their date of acquisition by the library. In addition, they analyzed titles in different LC classes separately, so LC class was also an implicit decision criterion in their study. These two criteria were used in addition to past-use statistics; when Fussler and Simon disallowed past-use statistics as criteria, they needed to combine even more criteria to obtain an adequate choice policy.

Before the advent of computerized transaction systems, research into these more complicated decision schemes was limited by the computational difficulty of managing all the necessary information about library books. Possible prediction methods had to be arrived at through ad hoc methods, and only small amounts of data could be gathered to evaluate their efficacy. Fussler and Simon’s study, probably the largest, examined a total of only 1,642 titles—all of them books—generating 1,601 transactions. Their titles were not uniformly distributed throughout the library system, but were instead concentrated in two subject areas, Economics and Teutonic Literature. In addition, the researchers found it difficult to estimate past-use information, since the sheet of paper containing book-use information, stored in the rear of each volume, was replaced after a few dozen entries. The computerization of library records, combined with computer algorithms that can be automatically trained to predict book use, makes possible a more complete examination of algorithms for predicting library book use, which we describe below.

With the availability of computerized databases of bibliographic and circulation information and more sophisticated computer resources, we can reproduce these previous studies using a much larger data set in a more exhaustive evaluation. The six decision criteria that we examine are shown in table 1. These criteria include all those examined by Fussler and Simon with the exception of acquisition date, which was not available for the Widener data. In addition, we consider CHECKOUT HISTORY AND COUNTRY OF PUBLICATION, which Fussler and Simon did not. We refer to two of these criteria—CHECKOUT HISTORY, measuring the number of past uses, and LAST USE, measuring the time since the last use—as past-use criteria, as they rely on the book’s circulation behavior in the past. In the simulation experiments reported on here, we use circulation records from July 1975 to June 1984 as the “past” for use by these criteria. Data from July 1984 to June 1993 are not available to the past-use statistics and are used to analyze decision trees. (See
The page contains a table and some text. The table is labeled as Table 1, which seems to be about criteria considered in models of predicting book use. The table includes columns for criterion, description, and example values. The text discusses the methodology of evaluating decision trees and compares the performance of various decision criteria. The figures illustrate the performance of single-criterion choice policies and detail for small off-site percentages.
The Library Quarterly

![Graph showing performance of decision trees](image)

**Fig. 4** — Performance of choice policies recommended by Fussler and Simon [3]. Policies using use records outperform those that do not in general, but not when less than 20 percent of books are in the depository. Adding LC CLASS does not always improve performance, perhaps because of overtraining.

Of the first two variants, the one taking advantage of past-use information is again better overall, but when few titles need to be put in the depository, it performs worse than the method based purely on bibliographic information. This makes sense, since in this low range all titles being considered will never have been checked out, and the last use policy will be picking titles basically at random. Surprisingly, using LC CLASS as an extra criterion can degrade performance, adding LC CLASS to the “no use records” policies reduces the EAR from 52 to 47 percent. It is possible that adding another criterion makes the algorithm prone to overtraining (See Sec IVC).

In summary, past-use statistics are the best single criterion for predicting book use, although contra previous studies, other criteria dominate when small percentages of books (less than about 18 percent) are stored off-site. The addition of extra criteria to past use can, to an extent, further improve predictive power, but care must be taken, as degradation can also result.

**III. The Methodology of Decision Trees**

The multicriterion algorithms of Fussler and Simon can be seen as variants of a general class of methods based on decision trees. A decision tree is a hierarchical structure for classifying objects, composed of nodes

![Diagram of decision tree](image)

**Fig. 5** — A simple decision tree

that correspond to primitive classification decisions. For the task at hand, the objects to be classified are titles and we wish to classify them in such a way that the classes are maximally predictive of their future use. The primitive classification decisions are simply the criteria that are available for classifying titles, as listed in table 1.

At the top of a decision tree (called, perversely, the root; decision trees grow down rather than up) is a node that specifies the main dividing criterion for subclassifying the titles. The dividing criterion might be, for instance, LANGUAGE. For each value of this criterion—English, Swahili, Chinese, and so forth—the node has a child node, which can be thought of as classifying further all the titles with the given language value. Associated with each node, in addition to a dividing criterion, is a set of titles. The root node contains all the titles, while child nodes contain those titles they inherit from their parent. In our example, the English child node inherits from its parent all titles written in English, while the Swahili node inherits titles in Swahili, and so on.

Each of these nodes in turn can have a dividing criterion and children of its own. In this way, the set of titles can be subclassified into finer and finer subgroups, where the nodes at the bottom of the tree, the leaves, constitute an exhaustive classification of all of the titles into disjoint classes. Figure 5 shows a simple decision tree. Titles are divided first on the basis of the language they are written in. Those written in English are further subdivided on the basis of their country of publication. (For purposes of exposition, we limit attention in the figure to a small subset of the possible values for these criteria.)

Decision policies treat each leaf as a unit, ranking each leaf on the
basis of its expected future use—or, rather, the average expected future use of titles in that leaf. The variation in prediction quality among decision trees comes from the choice of decision tree. The number of levels in the tree, the methodology for detecting and weeding out ineffective leaves, and even the order in which we choose the dividing criteria, can all affect the quality of prediction. By way of example, we have already seen the difference in performance using a zero-level tree (random) as opposed to a good one-level tree (last use).

A. Defining and Evaluating Choice Policies

A decision tree can be used as the basis of a choice policy by ordering the leaf nodes according to which classifications are expected to have the lowest hit rate. We calculate this expected hit rate for each node using data from 1984–93. (Recall that data from 1975–84 are reserved for the past-use statistics.) That is, each node is assigned a value based on the number of times titles in that node were checked out in the period 1984–93. We can, in this case, think of the data from 1984–93 as the "recent past" and data used by past-use statistics as stemming from the "distant past."

This ordering of nodes induces a corresponding ordering on the associated sets of titles. When we need to pick titles to place off-site, we start taking titles (in arbitrary order) from the lowest-ranked node's classification, moving on to higher-ranking nodes as the earlier nodes are emptied. If use patterns from the recent past, which we used in our ordering, hold into the future, the ordering we generate will be the best one.

This procedure is adequate for creating and using decision trees, but we need some more data in order to test their efficacy. One way to do this is to garner some "future" data and, as we move titles to the depository, track how many times the titles are checked out in the future. This technique allows us to calculate the hit rate at different percentages of off-site storage. In this way we can generate curves such as those in the previous figures.

This method can be improved on by using a different data set of titles to test the decision tree, which we call the testing set to distinguish it from the training set used to construct the structure and ordering of nodes in the decision tree. This way we are sure that we are evaluating the predictive power of the way the tree divides titles in general as opposed to its predictive power on the specific training set titles. We would use testing set circulation data from the recent past to calculate past-use statistics; this is necessary so that we can determine the appropriate node for each title of the testing set. Then we could use the data from the future to evaluate how well the tree predicts future use.

Unfortunately, we have no data from the future. However, since the data in the testing set are unrelated to the data in the training set, it is acceptable to move the entire time frame for the testing set backward. Therefore we can use the distant past for past-use information and the recent past to simulate the future. This technique will fail only if patterns of use change rapidly enough to make predictions based on the recent past (1984–93) invalid in the future. We do not expect this to be the case.

A few examples may clarify the process. The random choice policy is defined by a decision tree with a single node. Since no dividing goes on, all titles in the training set are placed in that one node as a single class. That node is the only leaf node, so that the sorting of the leaf nodes according to their recent past use is trivial. Next the node is cleared and replaced by titles in the testing set; again, since there is only one node all the titles are placed in it. These titles are then chosen in an arbitrary order to be moved to the depository. Since the books are chosen arbitrarily from the single group, hit rate goes up linearly with off-site percentage.

The single-criterion choice policies of figures 2 and 3 correspond to decision trees of a single level, with a root node that divides on the given criterion. For example, consider the last use single-criterion choice policy. The corresponding decision tree is depicted in figure 6. The root node is the only nonleaf node of the tree. The leaf nodes are sorted based on their performance in the recent past (undoubtedly with more recent last-use nodes sorted ahead of less recent ones). The nodes then cleared of titles and replaced with titles from the testing set. The distant past behavior of the testing set titles is used to divide the titles according to the last use dividing criterion. The titles from the testing set are then selected for off-site storage according to the sorting order previously determined. The recent past use data for the testing set are then used to determine the hit rate as a function of off-site percentage.

---

**Fig. 6**—A single-criterion decision tree that divides on the last use criterion.
The "no use records" Fussler and Simon policy of figure 4 corresponds to a decision tree where the root node divides on Language and the children of the root node divide on publication date. Figure 7 displays this decision tree.

B. Sampling Issues
At the time of this study, 2.2 million of the estimated 3.8 million Widener Library titles had been cataloged in Harvard University's online library computer system, having generated a total of six million transactions since July 1975. A campaign is in progress to computerize the rest of the titles, most of which have not generated a single transaction. The results we have obtained thus apply not to Widener as a whole but to some "sub-Widener" that excludes many relatively unpopular titles. However, the relative comparisons are still valid, assuming that one decision scheme would not benefit inordinately from the noncomputerized titles. This seems probable. Once the ongoing effort to computerize all titles has been completed, it should be possible to tailor the decision-making policy to the true population of Widener.

Because of computational limitations, we do not divide the entire collection into two data sets of 1.1 million titles each. Instead, we make the training and testing sets somewhat smaller, approximately eighty thousand titles. In subsampling the data, it is important to ascertain that a sufficiently large subsample is being used. Figures 8 and 9 show the performance of a simple last use decision tree on subsamples varying in size from five thousand to eighty thousand titles. Notice that the performance of the decision tree converges rapidly as the sample size increases, indicating that the results of our experiments, which use data sets of eighty thousand titles, should be applicable to larger and smaller data sets as well. Additional tests showed that the same trend holds for other decision trees, including the much more complicated ones discussed in future sections. Thus, a sample size of eighty thousand titles seems sufficient.

As another verification of the reliability of the EAR figures that we
calculate through simulations, we calculate the maximum error for the EAR value of each choice policy due to subsampling. Each choice policy is tested on eight different test sets. In the left-hand portions of the figures, a set of averaged curves is shown, one for each choice policy. In the right-hand portion of the figure, the mean EAR value is shown with error bars bracketing the 95 percent confidence interval. This serves to delimit the range that, with high probability, the EAR value would have fallen within if no subsampling had been performed. The right-hand policies are keyed to the left-hand legend with the bracketed numbers. In general, the intervals are quite small and serve to confirm the reliability of the differences among the choice policies.

IV. Designing Decision Trees

Once we characterize previous algorithms for predicting book use—from simple one-criterion tests to the more complicated tests developed by Fussler and Simon—as decision trees of one or two levels, it seems natural to look at even larger decision trees. After all, it surely cannot hurt to consider as much information about a title as possible before deciding whether to move it off-site.

Once a node in a decision tree divides on a given criterion, it makes no sense to divide again lower in the tree on that same criterion. Therefore, if we consider up to n criteria, a tree can include at most $n^2$ levels. Of course, we may choose to make a decision tree that is not so deep. Trees of maximum depth we call maximal trees. Since our study examines six different criteria (table 1), maximal trees have six levels between the root and each leaf node. To order to maximize the amount of information used in making storage choices, we should consider maximal decision trees.

Indeed, the question arises as to why previous researchers did not examine these more complicated methods themselves. The reason is undoubtedly one of impracticality. Some decision trees we consider have over forty-six thousand leaf nodes. That is, the eighty thousand titles have forty-six thousand distinct combinations of values for the six division criteria. Trees of this size are impossible to analyze without a computer. Even with the use of computers, computational limitations remain. Instead of creating a decision tree, which considers only a single classification criterion at a time, it might be better to calculate a regression analysis on the various criteria, determining not only how each criterion contributes to future book use but also how combinations of criteria contribute. Unfortunately, such analyses on data sets of greater than five thousand titles, even when calculated using only a limited sub-

set of the possible criteria, proved to be impossible given current technology. Nevertheless, it is certainly possible to evaluate maximal decision trees using the current technology.

A. Randomly Selected Maximal Decision Trees

All maximal trees differ only in the order of the dividing criteria; their leaf nodes classify the titles into the same disjoint classes. That is, the maximal trees have the same leaves but in a different order. Since our evaluation technique reorders the leaf nodes (based on predicted hit rate), all maximal trees are theoretically the same for the purposes of evaluation. It should not matter which we choose to evaluate.

The only exception to the equivalence of maximal trees would arise if the observed hit rate for two nodes is identical. In this case, some method for "breaking the tie" must be instituted so as to fully order the nodes. Unfortunately, such ties are quite common. For example, a maximal tree for one eighty-thousand-title sample had forty-six thousand leaf nodes but only 247 distinct past-use values to be used for sorting those nodes. There tend to be many ties because each leaf contains only a few titles. In contrast, a simple decision tree based only on Last Use might have 110 leaf nodes with ninety-three different values. Nodes with the same value look the same to any ordering algorithm, so a tie-breaking method must be invoked. An arbitrary decision here is not necessarily appropriate: it may be that one node is superior to the other in fact, but our sample size is too small to let us determine it.

The hierarchical structure of the decision tree turns out to be useful here. If two nodes are tied in the number of checkouts of their constituent titles, we can compare the nodes' parents instead. The parents have less rigorous dividing criteria and therefore contain more titles; the more titles a node has, the less likely its average hit rate is repeated in some other node. If another tie does arise, the remedy can be repeated, leading to examination of grandparents, and so forth. Only rarely will this technique fail to differentiate between two nodes, forcing us to pick one over the other arbitrarily. By using the parent nodes to perform the ordering, we can effectively increase the sample size at the cost of some specificity.

Thus, for large trees, the ordering of the decision criteria becomes important in determining the breaking of ties. (Order of division is also important when we smooth the maximal trees to eliminate problems of overtraining, as discussed in Sec. IV.C.) This can lead to varying performance among the different maximal trees.

In order to gauge the quality of maximal trees in general, rather than a specific maximal tree, we create each maximal tree randomly, as-
signing each node a random dividing category from the set of legitimate categories remaining to it. Each of the eight testing sets uses a different, randomly created maximal tree. This randomness explains the large confidence interval in the exhibited performance for maximal trees.

Figure 10 presents a comparison of such random maximal trees against the various trees proposed by Fussler and Simon. Surprisingly, the maximal tree is not the unequivocal top performer. For instance, Fussler and Simon’s past-use tree differentiated by LC class—which divides only on LC class, LAST USE, and PUBLICATION DATE—is significantly simpler than the maximal tree but performs almost as well, particularly when few titles need to be put in the depository. This does not mean, however, that maximal trees are inherently flawed; in fact, because of the large confidence interval it is hard to tell exactly how well maximal trees perform. We need a way of choosing the order of dividing on maximal trees in order to tighten the confidence interval, preferably improving mean performance at the same time.

B. ID3-Ordered Trees

It is possible to use an algorithm called ID3, developed by J. R. Quinlan [8], to find a good dividing order. For each node, the ID3 algorithm uses a heuristic to calculate the information gain inherent in dividing on different criteria and picks the criterion with the greatest gain. Information gain is highest when the output values of the children of a node are as different as possible. For instance, suppose the titles in a node are checked out on average 3.5 times in the recent past—the time period used to order the nodes. If we divide the node and create two children, one with an average checkout of 0 and the other with an average of 7, we have gained a lot of information because it is easy for us to decide which node to rank higher. On the other hand, if the children have average checkouts of 3.4 and 3.6, we have gained less information. For a mathematical description of how the ID3 algorithm decides on dividing criteria, see the appendix.

The ID3 algorithm is not guaranteed to give improved performance. However, we see in figure 11 that an ID3-ordered tree performs better than a maximal tree on average. The mean EAR value for the ID3-ordered tree lies just at the top of the 95 percent confidence interval for maximal trees, showing that the ID3 ordering is better than the vast majority of orderings for maximal trees. Equally important, the ID3-ordered tree has a much smaller confidence interval.

C. Overtraining and Smoothing

In addition to being subject to the tie-breaking problem described in the previous section, large decision trees are prone to overtraining: as the tree classifies titles into finer and finer classes, the ordering of the leaf nodes tailors itself to idiosyncrasies of the data set it is training on. Instead of capturing trends relating criteria to future use, the tree cap-
the small nodes. Good smoothing methods leave many nodes where they are needed to make fine distinctions between titles but eliminate nodes that contribute to overtraining without making any real contribution to categorization accuracy.

We explore the use of two smoothing methods on the maximal decision tree. The first uses a heuristic to find places where the tree is more developed than the data warrant. The heuristic looks at the variance of the hit rates for titles in each node. Nodes with small variances are themselves good predictors of book use and do not need to be divided further. The algorithm looks for such nodes, choosing the node whose variance is the smallest with respect to the average variance of its children. It then makes that node a leaf node, deleting its children and further descendants from the tree. The process is repeated, creating a succession of smaller and smaller trees. Each tree is evaluated and the tree with the best performance is chosen as the smoothed tree. Because of the way children are removed, we refer to this method of smoothing as pruning.

The second method of smoothing works in the opposite direction: instead of pruning the children of a given node, it folds a given node into its parent. We refer to this method as backing off. We back off a node if we doubt the reliability of its estimate of the future hit rate for that node. We determine reliability by looking at the number of circulation events that the titles classified by the node account for. In particular, we define a node's size as the number of titles in the node plus the number of past checkouts for all titles in the node. Nodes with small size are unreliable because their scant use history increases the variance of their future-use information. We ignore small nodes, considering their much larger parents in their stead. As in the pruning method, a succession of trees is created, each formed using a different maximal size for backing off. Again, the tree with the best performance is chosen as the smoothed tree.

Both smoothing algorithms generate a series of trees and require us to judge their performances. What data set can be used to make this judgment? We cannot use the training set, because the whole point of smoothing is to alleviate the tree's dependence on the training set. We cannot use the testing set for the same reason as we cannot train using the testing set: it would constitute cheating and cause the testing procedure to underestimate the true hit rate. We must instead use a third data set, which we will call the smoothing data set. To do this, we divide the training data set in two, using half of it as the new training set and the other half as the smoothing set.9

9. Another small change is required in our evaluation procedure. Recall that only leaf
We compare ID3-ordered trees produced by the two smoothing methods with the unsmoothed tree in figure 13. The backed-off tree, with an EAR of 73.1 percent, performs the best, while the pruned tree performs worse than the unpruned tree, leading us to question the efficacy of the pruning heuristic. As shown in figure 14, the backed-off, ID3-ordered tree performs consistently better than trees developed by Fussler and Simon and other researchers. The improvement is particularly striking when 20–40 percent of titles need to be moved off-site, a reasonable range for many research collections. In all ranges, however, the backed-off, ID3-ordered tree is the best decision tree we have studied.

D. Further Improvement
Although our best choice policy is a significant improvement over previous proposals, it is still far from matching the performance of the clairvoyant policy. Various possibilities might be entertained to further close this gap. First, the ID3 method has been surpassed in recent years by other algorithms for ordering decision trees that apply increasingly sophisticated statistical tests to the data. Unfortunately, these more sophisticated algorithms do not lend themselves to the large number of criterion values found in the library data set, and they are prohibitively slow as a result. If a more modern algorithm can be tailored to the library data, however, it may give improved performance.

Second, one could add more decision criteria. With good smoothing methods, it is possible to include many more than the six criteria we considered, while not overtraining the decision tree. The problem remains of finding other predictive criteria. Preliminary examination of the type of a title—monograph, serial, and so forth—indicates that this statistic does not improve the accuracy of prediction. Other criteria available in the Harvard bibliographic database, examining the author’s name, for instance, or whether the title includes illustrations, are even less likely to improve performance.

V. Conclusions
Given the importance of choosing a good decision tree to implement a choice policy for off-site storage, we explored several approaches for constructing decision trees. These allow us to say which combinations of criteria—of the ones we studied—best predict future use.

We follow most previous studies by endorsing past use as the best single predictor of future use. We do so, however, with reservations. When a large percentage of a library’s collection needs to be held off-
site, the best criterion is checkout history, the number of past circulations. However, when only a small percentage of a collection needs to be moved (less than 18 percent in our study), past use is less useful. This is because a large proportion of a collection may never have been checked out, and past-use statistics are unable to distinguish among the books in this subpopulation. Instead, language of publication and LC class seem to be the best criteria when few titles need to be put in the depository.

It is possible to combine the best of all worlds by using more than one criterion to predict future book use. The logical extension of this is to use all the criteria available in our prediction rules. Unfortunately, this causes decision trees to be too large for the data set, causing several problems. Tie breaking can be solved by picking the nodes of the tree carefully; computational heuristics such as ID3 can be used to try to pick the best ordering. Overtraining can be solved by smoothing, which shrinks the decision tree in places where the extra granularity is not needed. The smoothed, ID3-ordered, maximal decision tree convincingly outperforms any single-criterion decision tree and is the best method we have tested for predicting book use.

By way of illustration, if the Harvard College Library had implemented a last-use policy, as recommended by Fussler and Simon, to choose which 20 percent of its collection to move to the depository in 1985, they would have had to retrieve volumes from the depository about thirty-four thousand times per year. If they had, instead, used the smoothed, ID3-ordered, maximal tree, there would have been less than a fifth as many, only 6,200 hits per year. In comparison, a random choice policy would have resulted in sixty thousand hits per year, while a clairvoyant policy would have garnered zero hits.

Appendix

The ID3 Algorithm

The ID3 algorithm is used to choose a dividing criterion for a given node. By applying the ID3 algorithm to the root of a one-node decision tree we obtain a decision tree with a root and several children. The ID3 algorithm can be applied recursively to each of the children to create an entire decision tree, terminating when the algorithm determines there is no appropriate dividing criterion for any leaf node of the tree. The following formulas are taken from Quinlan's original paper [8].

Suppose we are considering leaf node $N$, which has $p_N$ titles. Let $q_i$ be the number of titles in $N$ that were checked out $i$ times in the "recent past." The information inherent in node $N$, $I(N)$, is defined to be

$$I(N) = - \sum_i \frac{q_i}{p_N} \log_2 \frac{q_i}{p_N}$$

This quantity is measured in bits, since it represents the number of computer bits needed to store the information in a node.

Suppose we tentatively choose criterion $C$ as a dividing criterion and divide $N$ based on criterion $C$. Call the children of $NN_1, \ldots, N_v$, where $v$ is the number of values for criterion $C$. Let $p_i$ be the number of titles $N_i$ inherits from $N$. The expected information to create the children is defined by

$$E(C) = \sum_{i=1}^{v} \frac{p_i}{p_N} I(N_i).$$

The information gain in dividing node $N$ on criterion $C$ is therefore

$$\text{gain}(N, C) = I(N) - E(C).$$

The use of the term "gain" is perhaps a bit misleading, because while it indeed measures the information gain of the children of $N$ over $N$ itself, it does not take into account the information required to make the division. This statistic can be expressed as

$$IV(N, C) = - \sum_{i=1}^{v} \frac{p_i}{p_N} \log_2 \frac{p_i}{p_N}.$$  

We wish to maximize the quantity

$$\text{gain}(N, C)/IV(N, C).$$

This gain ratio statistic has the advantage over the gain statistic in that it does not favor criteria that splinter the data into many criteria, which may make the gain quantity large due merely to the overwhelming magnitude of the summation limit. The gain ratio suffers from its own problem, however, in that it may inordinately favor criteria that have a near-zero value of $IV$. We therefore use a combination of the gain and gain ratio statistics in our final decision algorithm.

Suppose that there are $n$ possible criteria on which to divide node $N$. We choose the dividing criterion for node $N$ as follows. (1) Choose the $n/2$ criteria that have above-average gain for dividing on node $N$. (2) Discard those criteria that have 0 gain. If no criteria remain, do not divide node $N$. (3) Otherwise, output that criterion that maximizes the gain ratio

$$\text{gain}(N, C)/IV(N, C).$$
The gains and gain ratios for each criterion when calculated on the root node of a decision tree are summarized in Table A1. The results predicted by the ID3 algorithm should parallel those of the one-criterion decision trees (Fig. 2). For the most part, the ID3 algorithm does well, ranking check-out history and last use first, but it inaccurately claims country is the next most useful criterion. The algorithm is a heuristic and is not guaranteed to give optimal results.

REFERENCES


